We claim:

1. A method of scheduling central processing unit (CPU) usage by a given task comprising:

associating said given task with a top level class and a sub-class; and

determining a target CPU usage for said given task from a weight associated with said sub-class and a target CPU usage associated with said top level class.

2. The method of claim 1 further comprising:

determining an actual usage of said CPU by said given task in a first predetermined evaluation interval;

determining a penalty duration for said given task based on said actual usage and said target CPU usage for said given task; and

applying a penalty to said given task for said penalty duration during a second predetermined evaluation interval.

3. The method of claim 2 wherein said applying said penalty comprises demoting a scheduling priority associated with said given task.

4. The method of claim 2 wherein said penalty is applied continuously for said penalty duration.

5. The method of claim 2 wherein said penalty is applied during a plurality of periods over said second predetermined evaluation interval, such that a total duration of application of said penalty is equivalent to said penalty duration.

6. The method of claim 2 wherein said actual usage of said CPU by said given task in said first predetermined evaluation interval is a first actual usage and said penalty duration based on said first actual usage is a first penalty duration, said method further comprising:

determining a second actual usage of said CPU by said given task in said second predetermined evaluation interval;

determining a second penalty duration for said given task based on said second actual usage and said target CPU usage for said given task; and

applying said penalty to said given task for said second penalty duration during a third predetermined evaluation interval.

7. The method of claim 1 wherein said sub-class is associated with a parent class.

8. The method of claim 7 wherein said weight associated with said sub-class represents a relative share of a target CPU usage associated with said parent class.

9. The method of claim 8 wherein said sub-class is one of a plurality of sub-classes directly associated with said parent class and said determining said target CPU usage for said given task comprises:

forming a quotient by dividing said weight associated with said sub-class by a sum of weights associated with said plurality of sub-classes directly associated with said parent class; and

multiplying said target CPU usage associated with said parent class by said quotient.

10. The method of claim 8 wherein said top level class is said parent class of said sub-class.

11. The method of claim 8 wherein a further sub-class of said top level class is said parent class of said sub-class.

12. An apparatus for scheduling usage of a central processing unit (CPU) operable to:

associate a given task with a top level class and a sub-class; and

determine a target CPU usage for said given task from a weight associated with said sub-class and a target CPU usage associated with said top level class.

13. The apparatus of claim 12 further operable to:

determine an actual usage of said CPU by said given task in a first predetermined evaluation interval;

determine a penalty duration for said given task based on said actual usage and said target CPU usage for said given task; and

apply a penalty to said given task for said penalty duration during a second predetermined evaluation interval.

14. A computer readable medium containing computer-executable instructions that, when performed by an apparatus for scheduling usage of a central processing unit (CPU) in a kernel, cause said apparatus to:

associate a given task with a top level class and a sub-class; and

determine a target CPU usage for said given task from a weight associated with said sub-class and a target CPU usage associated with said top level class.

15. The computer readable medium of claim 14 wherein said computer-executable instructions further cause said apparatus to:

determine an actual usage of said CPU by said given task in a first predetermined evaluation interval;

determine a penalty duration for said given task based on said actual usage and said target CPU usage for said given task; and

apply a penalty to said given task for said penalty duration during a second predetermined evaluation interval.